יוסי חזן **עבודת בית עבור הנדסאי תוכנה** נתונה הטבלה הבאה:

רשום ליד כל פקודה האם היא חוקית או לא,אם היא חוקית רשום מה היא מבצעת ואת התוצאות,ואם היא לא חוקית רשום מדוע.

|  |  |
| --- | --- |
| MOV DX,1234H | חוקי , אופרנד היעד בגודל 16 ביט ומקבל נתון בגודל 16 ביט |
| MOV AX,234H | חוקי , אופרנד היעד בגודל 16 ביט ומקבל נתון בגודל 12 ביט |
| MOV 5,AL | לא חוקי , אופרנד היעד הוא מספר ולא אוגר |
| MOV AL,546H | לא חוקי , אופרנד היעד בגודל 8 ביט ומקבל נתון בגודל 12 ביט |
| MOV AL,AH | חוקי , אופרנד היעד ואופרנד המקור שניהם מכילים נתון בגודל 8 ביט |
| MOV AX,BX | חוקי אופרנד היעד ואופרנד המקור שניהם בגודל 16 ביט |
| MOV AX,AL | חוקי , אופרנד היעד בגודל 16 ביט ואילו הוא מקבל נתון בגודל 8 ביט |
| MOV AX,TABLE | לא חוקי, |
| MOV TABLE,AX | ----- |
| MOV ES:[BX],AX | ---- |
| MOV BL,AH | חוקי , אופרנד היעד ואופרנד המקור שניהם מכילים נתון בגודל 8 ביט |
| MOV CL,-30 | חוקי , |
| MOV BYTE PTR[BX],35H | חוקי |
| MOV [BX]AL | חוקי |
| MOV [SI],47H | לא חוקי |
| MOV BYTE PTR[SI],56H | חוקי |
| MOV [BX],256 | לא חוקי |
| MOV [300],5 | לא חוקי |
| MOV BYTE PTR DS:[300],5 | חוקי |
| MOV [BX],BL | חוקי |
|  | הערה:  לאוגרי סגמנט ( כמו למשל ( DSלא ניתן לשים ישירות מספר קבוע לכן צריך להשתמש באוגר מתווך |
| MOV AX,5 MOV DS,AX | חוקי |
| MOV DS,DATA |  |
| MOV DS,5 | לא חוקי |
| MOV DX,OFFSET MSG |  |
| MOV BX,1000H  MOV AL,88H  MOV [BX],AL |  |
| MOV AX,BALANCE[SI] |  |
| MOV AX,8[BX] |  |
| MOV AX,[BX][SI] |  |
| MOV AX,4[BX][SI] |  |
| MOV BL,A2[DI] |  |
| MOV AA[2],BB[1] |  |
| MOV CS,A1 |  |
| MOV BL,A3[CX] |  |
| MOV AX,[DI] | חוקי , אופרנד המקור מצביע על נתון בזיכרון שיכנס לאופרנד היעד |
| MOV [DI],AX |  |
| MOV [SI],A2 |  |

|  |  |
| --- | --- |
| MOV BX,OFFSET A3[SI] |  |
| MOV BYTE PTR A1[DI+SI],BL |  |
| רשום פקודה אחת השקולה לצמד הפקודות הבא:  MOV BX,100  MOV AX,[BX] |  |
| MOV AX,11010001B |  |
| MOV DL,[AX] |  |
| MOV AL,0 |  |
| MOV CS,A2[0] |  |
| MOV [DI],[SI] |  |
| INC [BX] |  |
| INC DX |  |
| INC TAVIT | חוקי |
| DEC DL |  |
| DEC SI |  |
| DEC [SI] |  |
| ADD AH,8 |  |
| ADD DL,[BX] |  |
| ADD [BX],92H |  |
| ADD WORD PTR[BX],92H |  |
| ADD AL,AL |  |
| ADD AL,BYTE PTR A2 |  |
| ADD A3[DI],0AH |  |
| ADD AL,A3+[SI] |  |
| ADD AL,DX |  |
| SUB AX,20H |  |
| SUB AL,AL |  |
| SUB A2[BX+DI],DH |  |
| SUB [BX],BL |  |
| SUB DX,A2+BX |  |
| CMP CL,9 |  |
| CMP AX,CX |  |
| CMP [BX],CL |  |
| CMP CL,AX |  |
| CMP [DI],5 |  |
| CMP [SI],[DI] |  |
|  | עבור ההוראות יש לרשום אם חוקי או לא ולמה זה שקול |
| MOV AX,100H MOD 17 |  |
| MOV BX,0EFH SHL 2 |  |
| MOV DL,3EH OR 0FEH |  |
| MOV DL, 5EH GE 6EH |  |
|  | רשום עבור ההגדרות הבאות אם חוקי או לא ואם כן שרטט את מצב הזיכרון |
| DB 42H |  |
| DB ? |  |
| DB 'P' |  |
| DB 'ISRAEL' |  |
| DW 20,30,40,50,60 |  |
| DB 10H DUP ('A) |  |
| DB 10H DUP (?) |  |
| DW 35H DUP(35DH) |  |
| STRING DB 3 DUP ('LA') |  |
|  | הערה:  ניתן לקנן את DUP עד 8 פעמים לדוגמא:  STAM DB 4 DUP(3 DUP (2 DUP(?))) תגרום להקצאת 42 בתים ללא ערך תחילי |
| A DB 2 DUP ('A1',281) |  |
| B DB 2\*3 DUP (2,-2) |  |
| C DW (A2B1H,125) |  |
| ARR1 DB 1000 DUP (?) |  |
| ARR1 DB 100 DUP (0) |  |
| ARR DB 100 DUP (1,2,3) |  |
| VAR DW 2,1,1234H |  |
| VAR DD 0FFFFFFFFH |  |
| VAR2 DQ 1289180H |  |
| VAR3 DT 123456H |  |
| STR DW 'ABC' |  |
| STR DQ 'ABC' |  |
| STR DB 'ABC' |  |
| A3 DW 12/5 DUP ('1',23) |  |
| A4 DB AH |  |
| A1 DW 3\*17H |  |
| A2 DB 3\*90 |  |
| A3 DW 2 DUP (39,-39) |  |
| A4 DB '-2',-2 |  |
| A2 DW 37,-37 |  |
| A3 DB 2 DUP (17,12H\*12H) |  |
| A4 DB DDH |  |
| A2 DW '-123' |  |
| A1 DB 12\*15H |  |
| A4 DB A2H |  |
| A1 DB 12 MOD 5 DUP (10H\*15) |  |